INSTITUTE OF PHYSICS PUBLISHING CLASSICAL AND QUANTUM GRAVITY

Class. Quantum Grav. **22** (2005) S1189–S1196 doi:10.1088/0264-9381/22/18/S33

A simple line detection algorithm applied to Virgo data

F Acernese[1,2,3], P Amico[4,5], M Al-Shourbagy[6,7], S Aoudia[8], S Avino[1,2,3], D Babusci[9], G Ballardin[10], R Barillé[10], F Barone[1,2,3], L Barsotti[6,7], M Barsuglia[11,12], F Beauville[13], M A Bizouard[11,12], C Boccara[14], F Bondu[8], L Bosi[4,5], C Bradaschia[6,7], S Braccini[6,7], A Brillet[8], V Brisson[11,12], L Brocco[15,16], D Buskulic[13], G Calamai[17,18,19,20], E Calloni[1,2,3], E Campagna[17,18,19,20], F Cavalier[11,12], R Cavalieri[10], G Cella[6,7], E Chassande-Mottin[8], C Corda[6,7], A C Clapson[11,12], F Cleva[8], J P Coulon[8], E Cuoco[10], V Dattilo[10], M Davier[11,12], R De Rosa[1,2,3], L Di Fiore[1,2,3], A Di Virgilio[6,7], B Dujardin[8], A Eleuteri[1,2,3], D Enard[10], I Ferrante[6,7], F Fidecaro[6,7], I Fiori[6,7], R Flaminio[10,13], J D Fournier[8], S Frasca[15,16], F Frasconi[6,7,10], A Freise[10], L Gammaitoni[4,5], A Gennai[6,7], A Giazotto[6,7], G Giordano[9], L Giordano[1,2,3], R Gouaty[13], D Grosjean[13], G Guidi[17,18,19,20], S Hebri[10], H Heitmann[8], P Hello[11,12], L Holloway[10], S Kreckelbergh[11,12], P La Penna[10], V Loriette[14], M Loupias[10], G Losurdo[17,18,19,20], J M Mackowski[21], E Majorana[15,16], C N Man[8], M Mantovani[6,7], F Marchesoni[4,5], E Marchetti[17,18,19,20], F Marion[13], J Marque[10], F Martelli[17,18,19,20], A Masserot[13], M Mazzoni[17,18,19,20], L Milano[1,2,3], C Moins[10], J Moreau[14], N Morgado[21], B Mours[13], A Pai[15,16], C Palomba[15,16], F Paoletti[6,7,10], S Pardi[1,2,3], A Pasqualetti[10], R Passaquieti[6,7], D Passuello[6,7], B Perniola[17,18,19,20], F Piergiovanni[17,18,19,20], L Pinard[21], R Poggiani[6,7], M Punturo[4,5], P Puppo[15,16], K Qipiani[1,2,3], P Rapagnani[15,16], V Reita[14], A Remillieux[21], F Ricci[15,16], I Ricciardi[1,2,3], P Ruggi[10], G Russo[1,2,3], S Solimeno[1,2,3], A Spallicci[8], R Stanga[17,18,19,20], R Taddei[10], D Tombolato[13], M Tonelli[6,7], A Toncelli[6,7], E Tournefier[13], F Travasso[4,5], G Vajente[6,7], D Verkindt[13], F Vetrano[17,18,19,20], A Viceré[17,18,19,20], J Y Vinet[8], H Vocca[4,5], M Yvert[13] and Z Zhang[10]

[1] INFN Sez. di Napoli, Italy
[2] Università di Napoli 'Federico II', 80126 Napoli, Italy
[3] Università di Salerno, 84084 Fisciano (SA), Italy
[4] INFN Sez. di Perugia, 06123 Perugia, Italy
[5] Università di Perugia, 06123 Perugia, Italy
[6] INFN Sez. di Pisa, 56127 Pisa, Italy
[7] Università di Pisa, 56127 Pisa, Italy
[8] Observatoire de la Côte d'Azur, 06034 Nice, France
[9] INFN, Laboratori Nazionali di Frascati, 00044 Frascati (RM), Italy
[10] European Gravitational Observatory (EGO), 56021 Cascina (PI), Italy
[11] Laboratoire de l'Accélérateur Linéaire, France
[12] CNRS-IN2P3 and Université de Paris Sud, 91898 Orsay, France
[13] Laboratoire d'Annecy-le-Vieux de Physique des Particules, 74941 Annecy-le-Vieux, France
[14] ESPCI, 75005 Paris, France
[15] INFN Sez. di Roma, 00185 Roma, Italy
[16] Università di Roma 'La Sapienza', 00185 Roma, Italy
[17] INFN Sez. di Firenze/Urbino, Urbino, Italy
[18] Università di Firenze, 50019 Sesto Fiorentino, Italy

[19] Osservatorio Astrofisico di Arcetri, 51125 Firenze, Italy
[20] Università di Urbino, 61019 Urbino, Italy
[21] LMA, 69622 Villeurbanne, Lyon, France

E-mail: Eric.Chassande-Mottin@obs-nice.fr and Irene.Fiori@pi.infn.it

Received 29 March 2005, in final form 21 June 2005
Published 6 September 2005
Online at stacks.iop.org/CQG/22/S1189

Abstract

We propose a new method for the detection of spectral lines in random noise. It mimics the processing scheme of matching filtering, i.e., a whitening procedure combined with the measurement of the correlation between the data and a template. Thanks to the original noise spectrum estimate used in the whitening procedure, the algorithm can easily be tuned to various types of noise. It can thus be applied to the data taken from a wide class of sensors. This versatility and its small computational cost make this method particularly well suited for real-time monitoring in gravitational wave experiments. We show the results of its application to Virgo C4 commissioning data.

PACS numbers: 04.80.Nn, 07.05.Kf

(Some figures in this article are in colour only in the electronic version)

1. Introduction

Persistent and narrow spectral peaks, known as *lines*, are a typical feature of the data from gravitational wave interferometric detectors (ITF). Lines can originate from the ITF functioning (e.g., mirror or suspension resonant modes) and operation (e.g., calibration lines) or from environmental perturbations (e.g., vacuum pumps). Detecting the presence of lines in the ITF readout and control channels, and tracking their evolution is very important for the characterization and monitoring of the performance of ITFs. It can help ‘noise hunting’ by giving hints such as: (i) the appearance of lines may indicate the onset of a new noise source or point to a detector malfunctioning which, for example, is causing the excitation of some ITF internal resonance (e.g., a mechanical resonance of a mirror suspension); (ii) the simultaneous presence of the same line in different channels suggests a possible coupling and (iii) the similar drifting of some lines indicates their common origin.

In this paper, we present a simple and versatile line detection tool particularly well suited for real-time monitoring. Its processing scheme mimics the one of matched filtering in the sense that it combines a whitening procedure and the measurement of the correlation between the data and a template. A key feature is the *original* background noise spectrum estimate incorporated into the whitening step. This estimate can easily be tuned to the different noises observed in the various channels we monitor, thus providing flexibility to the whole procedure.

Optimal strategies can be designed for the detection of sinusoidal signals in random noise [1]. However, they are computationally expensive and thus not very well suited for monitoring purposes. In addition, the amplitudes of the lines we are dealing with are usually large as compared to the noise level. A statistic which is sub-optimal, but computationally acceptable, suffices in this context. Contrary to [2, 3], we are not interested here in the removal of the line once detected. Although estimates of the line characteristics (such as the central frequency, amplitude and width) can be obtained from the proposed tool, we do not discuss this point here and concentrate on the detection issue. Our line search algorithm shares similarities with *LineMon* which is one of the monitors of the GEO600 detector characterization system [4]. However, the two approaches differ in several aspects, in particular in the crucial point of their background noise estimate.

In section 2, we give a general description of the line detection problem and point out its main difficulties. In section 3, we detail the proposed line search method. In section 4, we give several rules of thumb for adequately setting the free parameters of the line search. In section 5, we present the results of its application to Virgo C4 commissioning data.

2. A tricky detection problem

ITFs are sophisticated apparatus controlled by a complex network of feedback loops. One way of locating the origin of a line in such a system is to check for the coincident occurrence of the line in channels including the ITF readouts, the ITF controls and the environmental sensors.

The line search has thus to be applied to many different channels. Generally, the spectral lines are superimposed onto a broadband random noise, hereafter referred to as 'background' noise. The power spectral density (PSD) of the background noise can have very different shapes. In most cases, we only have a rough idea of the PSD shape which may also change with time. Therefore, the problem to address is the statistical detection of a signal (lines) in a random noise of unknown PSD which thus needs to be estimated from the data. This turns out to be difficult because lines and background noise mix at all times, i.e., there is no 'background noise only' data. The proposed method tackles this difficulty.

3. A simple line detection algorithm

The basic ingredient of the proposed method is the *spectrogram*. Let us assume that the data $x(t + t_s n)$, $n \geqslant 0$, are sampled at Nyquist rate $f_s = 1/t_s$ and collected during the time period from t to $t + \mathcal{T}$. We divide this time period into $\mathcal{N}$ non-overlapping intervals of equal duration $\mathcal{T}/\mathcal{N} = t_s N$ (which thus contain the same number of samples N). The spectrogram $S(t, f)$ is defined for $f \in [0, f_s/2]$ by [5]

$$S^2(t, f) \equiv \frac{1}{\mathcal{N}} \sum_{k=0}^{\mathcal{N}-1} \left| t_s \sum_{j=kN}^{(k+1)N-1} x(t + t_s j)\, h(t_k - t_s j)\, \mathrm{e}^{-2\pi \mathrm{i}\, t_s j f} \right|^2,$$

where $t_k \equiv t_s(kN + (N - 1)/2)$ is the centre of the kth interval and $h(t)$ is a window function (e.g., Hanning type) centred around $t = 0$ and scaled to unit L^2 norm.

Matched filtering [6] is an efficient method for detecting deterministic signals in random noise. It can be viewed as a two-step process: a whitening of the data followed by a scalar product with a template. The line search algorithm mimics this structure. We assume that $S(t, f)$ has been computed for some given time t and we now detail the detection procedure.

Step 1. Background PSD estimate

We use $S(t, f)$ to get a robust estimate of the background PSD: (1) the (positive) frequency axis $f \in [0, f_s/2]$ is tiled into intervals. Their size is chosen sufficiently small that each of them contains only a few lines ($\lesssim 5$) and that the background PSD can be considered almost linear within the interval. (2) In each interval, we remove the N_q points with the largest amplitude (these 'outliers' essentially correspond to the few frequency peaks belonging to the interval). (3) We make a least mean square linear fit of the remaining spectrogram data points. The collection of the fits performed in all frequency intervals yields the estimate of the background noise PSD, $\hat{S}(t, f)$. Clearly, step (2) prevents the lines (of large amplitude) biassing this estimate.

Step 2. Whitening and scalar product with template

Roughly speaking, the lines can be described by the general model $a(t)\cos\varphi(t)$ where the envelope $a(t)$ is slowly varying and the phase $\varphi(t)$ is well approximated by a linear function for time periods of duration not smaller than $\mathcal{T}$.

The template matching to a line of duration $\mathcal{T}$ with a constant envelope $a(t) = C^{st}$ and a constant frequency $f_0 = (2\pi)^{-1}\,\mathrm{d}\varphi/\mathrm{d}t$ is a 'simple' cosine function of the same duration. The FFT (of time base $\mathcal{T}$) does exactly the scalar product with this template. The above definition of the spectrogram is not identical to this (because of the division of $\mathcal{T}$ into intervals and the modulus averaging), but it is similar. This suggests using $S(t, f)$ in place of the exact template match. We consider the following statistic $W(t, f) \equiv S(t, f)/\hat{S}(t, f)$ which includes both whitening and template match.

Step 3. Detection and post-processing

The line detection is then made by comparing the statistic to a threshold: $W(t, f) > \eta$.

Once the three steps are completed, we restart the procedure from step 1 to process the spectrogram computed over the next data chunk.

4. Choosing the parameters

Some tuning of the free parameters is required to ensure that the procedure works properly. There is a total of six free parameters. The computation of $S(t, f)$ uses two of them: the total observation time $\mathcal{T}$ (typically $\mathcal{T} \approx 300$ s) and the FFT time base $N \equiv \mathcal{T} f_s/\mathcal{N}$ (set to a power of 2 for efficiency). The latter sets the frequency resolution (the orders of magnitude are $f_s \approx 10$ kHz and we choose $N \sim 10^5$ data points corresponding to a frequency resolution of 100 mHz). The former fixes the number of averaged FFTs and consequently determines the amplitude of the fluctuations of the spectrogram S around its mean, which scales with $1/\mathcal{N}$.

The PSD fit requires a value for N_q. A good value is given by the mean number of lines per frequency interval times the peak average width (at half height, expressed in bins).

The threshold η determines the minimum detectable signal-to-noise ratio (or, more specifically, the ratio of the line amplitude to the neighbouring background noise level). As mentioned in [4], if we assume that the background noise is Gaussian and that the bias and variance of its PSD estimate $\hat{S}$ are small, the whitened spectrum W follows a Gamma distribution. With this approximation, the threshold can be explicitly related to the rate of false alarms. Typically, we choose $\eta \simeq 3$–4.

The two remaining parameters are related to the tiling of the frequency axis mentioned in section 3. We discuss the question of choosing their value in the following two sections.

4.1. Tiling types

The estimate of the background noise in step 1 of our algorithm relies on a tiling of the frequency axis. In principle, any tiling method could be used, but for practical reasons (simple coding) we restrict the choice to those described in this section.

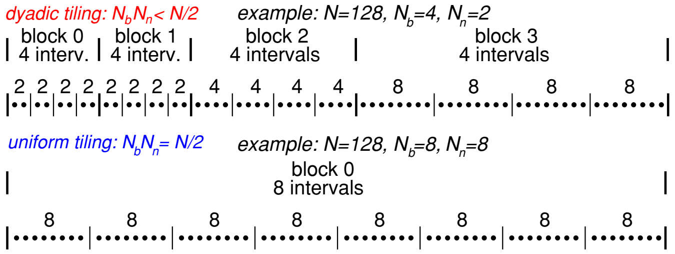

Figure 1. Sketch view of two examples of frequency axis tiling.

The FFT algorithm samples the frequency axis uniformly according to $f_k = f_s k/N$ for $k = 0, \ldots, N/2$. We propose to divide this discrete axis into non-overlapping intervals. These intervals are organized by successive blocks of intervals of equal size. Each block is identified by the index $j \in \{-1, 0, \ldots, j_{\max}\}$ and it is made of N_b contiguous intervals of size M_j. Each interval $\mathcal{I}_{i,j}$ is labelled by $i \in \{0, \ldots, N_b - 1\}$ and the block index j. We set $\mathcal{I}_{i,j} = \{k_{i,j}, \ldots, k_{i,j} + M_j - 1\}$ with $k_{i,-1} = iN_n$ and $M_{-1} = N_n$ for $j = -1$, and $k_{i,j} = 2^j(N_b + i)N_n$ and $M_j = 2^j N_n$ for $j \geqslant 0$. Therefore, the tiling is completely determined by the two parameters N_n and N_b, respectively, the smallest interval size (in bins) and the number of intervals within a block. Since N is a power of 2, N_n and N_b must also be so, and we have the condition $N_n N_b \leqslant N/2$. We also have $j_{\max} = \log_2(N/(2N_n N_b)) - 1$.

Two types of tiling are thus defined, depending on the product $N_n N_b$, as exemplified in figure 1. If $N_n N_b < N/2$, we obtain a *dyadic* tiling for which the size of the intervals is increasing with frequency. If instead $N_b N_n = N/2$, the frequency axis is tiled *uniformly* (all intervals are of size N_n).

4.2. Match the tiling to the data

The performance of the algorithm depends critically on the accuracy of the estimate $\hat{S}(t, f)$ of the background noise PSD, which itself depends on the selected tiling. The tiling choice results from a trade-off balancing the two following competing factors: long frequency intervals provide a better fit (because of the larger number of data points), whereas short intervals allow the better following of curved (non-flat) PSD.

The main readout channel of the ITF, i.e., the dark fringe (DF) signal, is of special importance, and deserves specific attention. The noise PSD observed in the DF signal has a characteristic shape: it is curved at low frequency (where the number of lines is also larger) and it becomes almost flat at high frequencies. Consequently, we need small intervals at low frequencies and large intervals at high frequencies. Dyadic tiling is thus preferred for the DF channel as illustrated in figure 2.

A different case is exemplified by a signal from a seismometer in figure 3. The PSD of the background noise for such a signal is much flatter than DF (compare the respective dynamical ranges) and its curvature is not localized specifically at low frequency. The uniform tiling is thus the right choice.

In summary, the tiling type should be chosen according to the general shape of the background PSD. However, for a given tiling type (dyadic or uniform), the size of the typical interval is also important. The fit made with intervals of small sizes is able to follow PSDs which are curved locally as illustrated in figure 4. If the size is too large, $\hat{S}(t, f)$ may not fit the data well and this causes an increase of the rate of false alarms.

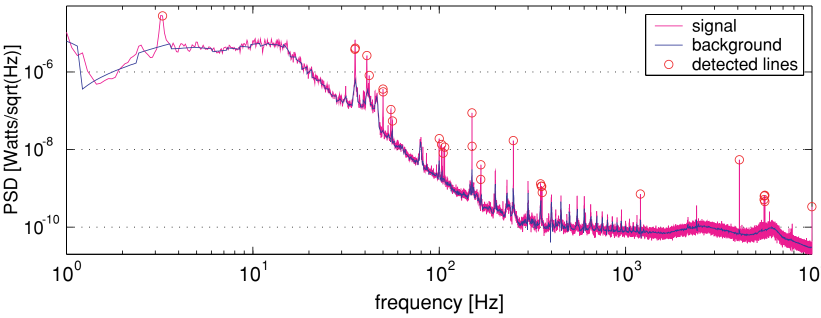

Figure 2. Result of the line search on the dark fringe photo-diode signal of Virgo C4 commissioning run ($f_s = 20$ kHz). The parameter set is $\mathcal{N} = 20$, $N = 262\,144$, i.e., total observation time $\mathcal{T} \simeq 260$ s, $N_b = 8$, $N_n = 64$, i.e., dyadic tiling, $N_q = 32$ and the detection threshold $\eta = 4$.

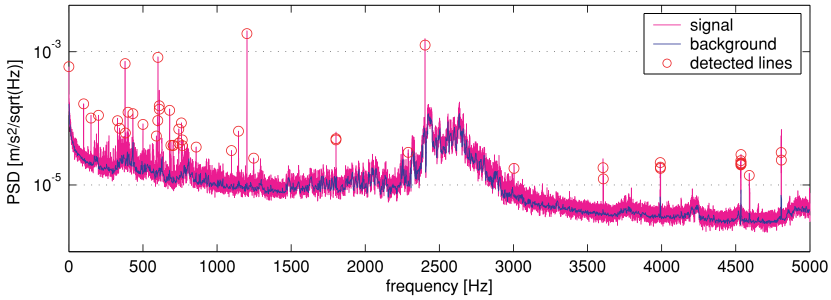

Figure 3. Result of the line search on one seismometer signal ($f_s = 10$ kHz). The parameter set is $\mathcal{N} = 20$, $N = 131\,072$, i.e., total observation time $\mathcal{T} \simeq 260$ s, $N_b = 1024$, $N_n = 64$, i.e., uniform tiling $N_q = 32$ and the detection threshold $\eta = 4$.

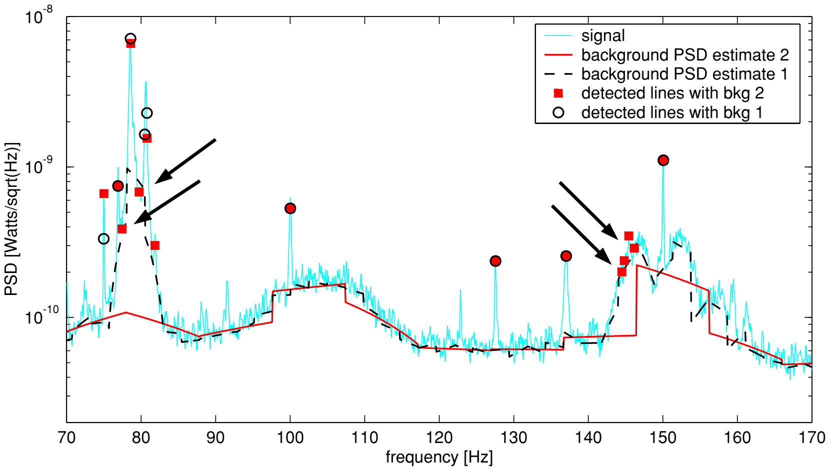

Figure 4. Line search on the photo-diode signal used in C4 to control the beam splitter mirror position. Two choices of the tiling size are made (1) $N = 262\,144$, $N_b = 32$, $N_n = 32$, $N_q = 16$, $\eta = 3$ (dashed line and empty circles); (2) $N = 262\,144$, $N_b = 32$, $N_n = 128$, $N_q = 64$, $\eta = 3$ (continuous line and filled squares). The coarser choice (2) has difficulty in following the curved shape of the background PSD. This causes false alarms which are indicated by arrows.

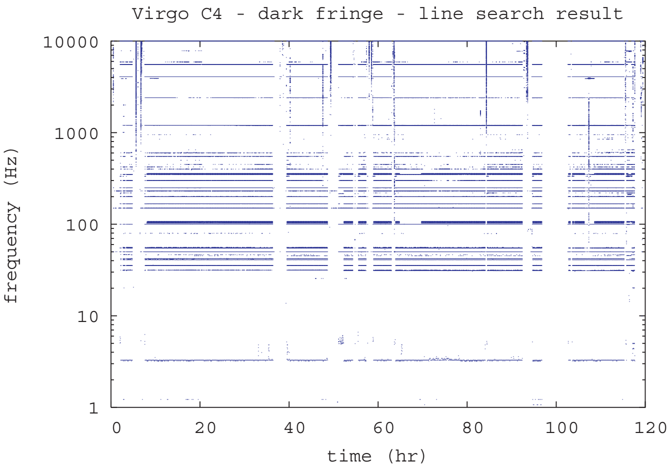

Figure 5. Lines detected in the Virgo DF during the entire C4 run (1 blue dot = 1 detection).

Table 1. Subset of 21 lines detected in Virgo C4 DF which have been identified (acronyms: IB, input bench; NI, north input; WI, west input; NE, north end; WE, west end).

Frequency (Hz)	Explanation	Frequency (Hz)	Explanation
0.4	Mirror suspension θ_x mode	353.0	Calibration line
3.2	Mirror suspension θ_x mode	355.0	Calibration line
31.5	IB suspension vertical mode	357.0	Calibration line
35.2	IB suspension θ_x mode	600.8	IB tower vacuum pump
40.9	IB suspension θ_z mode	1201.6	IB tower vacuum pump (1st harmonic)
55.0	IB wires violin mode	2402.4	IB tower vacuum pump (2nd harmonic)
56.1	IB wires violin mode	5543.9	NE mirror 1st symmetric mode
103.0	Calibration line	5544.8	WE mirror 1st symmetric mode
105.0	Calibration line	5583.9	WI mirror 1st symmetric mode
107.0	Calibration line	5585.8	NI mirror 1st symmetric mode
230.8	Water chiller pump		

5. Line monitoring in Virgo data

We used the proposed algorithm to monitor the lines of the Virgo DF in the C4 commissioning run data (a total of 120 h). The line search was applied over $\mathcal{T} \simeq 260$ s long data chunks with a FFT time base $Nt_s \simeq 13$ s (which corresponds to $N = 262\,144$ and a frequency resolution $\simeq$0.07 Hz). We chose the tiling parameters $N_b = 8$ and $N_n = 64$ (dyadic tiling), a rejection parameter $N_q = 32$ and a threshold $\eta = 4$.

A number of 40 lines were detected during the entire run. (This number is almost constant when ITF is in stable operation.) An overview is presented in the time–frequency diagram in figure 5 where one dot is one detected line. Isolated dots are probably false alarms, vertical stripes correspond to ITF losses of lock, blank regions correspond to unlocked periods. We see that some lines appear non-stationary.

We extracted a subset of 38 persistent lines, i.e., lasting at least 3% of the whole run duration. Among these 11 are 50 Hz harmonics, 21 have been associated with mirror internal modes, mechanical resonances of the mirror suspensions, vibrations of vacuum and water pumps and 6 have not been identified yet. Identified lines are listed in table 1.

6. Concluding remarks

We propose a robust and versatile line detection algorithm and we demonstrate its validity on Virgo C4 commissioning data.

The computational cost of the algorithm can be estimated. The computation of the spectrogram requires $\mathcal{N}$ FFTs of time base N, resulting in a cost increasing as $\mathcal{N} N \log_2 N$. The background PSD estimate (step 1) requires sorting all frequency intervals. Our implementation uses the `quick sort` [7] algorithm yielding a cost for step 1 scaling as $N/2(\log_2 N - \log_2 N_b)$ in the worst case. The rest of the calculation scales linearly with N. The total cost is thus dominated by the computation of the spectrogram. This is reasonably cheap and the processing pipeline runs in real time on a standard computer.

The algorithm is currently used for the detection of lines in the Virgo DF channel. The list of detected lines is made available and archived in the Virgo monitoring web pages [8]. The idea is to provide processed information to speed up the noise hunting investigations, for instance when tracking the history (appearance/disappearance) of a given line over long periods.

We are studying the possibility of feeding a database with the list of lines detected in the DF and in a selection of auxiliary channels. This would allow us to investigate the archives more easily and systematically through specific queries such as (i) searching for the coincidence of the same frequencies in two or more channels, (ii) the search for harmonics of a fundamental frequency and (iii) the identification of lines through the comparison with a list of known ones.

References

[1] Allen B, Papa M A and Schutz B F 2002 Optimal strategies for sinusoidal signal detection *Phys. Rev.* D **66** 102003

[2] Searle A C, Scott S M and McClelland D E 2003 Spectral line removal in the LIGO Data Analysis System (LDAS) *Class. Quantum Grav.* **20** S721

[3] Sintes A M and Schutz B F 1998 Coherent line removal: filtering out harmonically related line interference from experimental data, with application to gravitational wave detectors *Phys. Rev.* D **58** 122003

[4] Balasubramanian R, Babak S, Churches D and Cokelaer T 2005 GEO600 online detector characterization system *Preprint* gr-qc/0504140

[5] Guidi G M and Chassande-Mottin É (The Virgo Collaboration) 2003 Data analysis methods for non-Gaussian, non-stationary and non-linear features and their application to Virgo *Class. Quantum Grav.* **20** S915–24

[6] Kay S 1998 *Fundamentals of Statistical Signal Processing: Detection Theory* (Englewood Cliffs, NJ: Prentice-Hall)

[7] Press W H *et al* 1992 *Numerical Recipes in C* 2nd edn (Cambridge: Cambridge University Press)

[8] Virgo commissioning web pages http://wwwcascina.virgo.infn.it/commissioning/